



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/676,422	09/29/2000	Douglas N. Kimelman	YOR920000293US1	5708

33233 7590 12/15/2003

LAW OFFICE OF CHARLES W. PETERSON, JR.  
P.O. BOX 710627  
OAK HILL, VA 20171

EXAMINER

ALI, SYED J

ART UNIT PAPER NUMBER

2127

DATE MAILED: 12/15/2003

7

Please find below and/or attached an Office communication concerning this application or proceeding.

✓

# Office Action Summary

Application No.

09/676,422

Applicant(s)

KIMELMAN ET AL.

Examiner

Syed J Ali

Art Unit

2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 29 September 2000.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-28 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-28 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. §§ 119 and 120

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
a) ☐ All b) ☐ Some \* c) ☐ None of:  
1. ☐ Certified copies of the priority documents have been received.  
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).  
\* See the attached detailed Office action for a list of the certified copies not received.
- 13) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application) since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.  
a) ☐ The translation of the foreign language provisional application has been received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

## Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☒ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) 3.
- 4) ☐ Interview Summary (PTO-413) Paper No(s). \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other:

## DETAILED ACTION

### *Claim Rejections - 35 USC § 103*

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1-4, 9-15, 21-23, and 27-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunt (USPN 6,629,123) in view of French et al. (USPN 6,266,053) (hereinafter French).

As per claim 1, Hunt discloses a task management method for determining optimal placement of task components, said method comprising:

- a) generating a communication graph representative of a task (col. 23 lines 13-23, “the application units and inter-unit communication form a commodity flow network”, wherein the application units are components of a task or an application program, and the inter-unit communication provides information pertaining to the weighting of edges);
- c) determining a min cut for the communication graph (col. 24 lines 8-28, “the algorithm to map a client-server distributed partitioning problem onto the MIN-CUT problem is as follows”, wherein the algorithm for determining the minimum cut of the graph is disclosed); and

Art Unit: 2127

d) placing task components responsive to said min cut determined for the communication graph (col. 23 lines 13-23, "After all data has been gathered, it is the optimization algorithm that decides where application units will be placed on the network", wherein the optimization algorithm is a minimum cut algorithm and finds the paths of minimal communication costs).

French discloses the following limitations not shown by Hunt, specifically:

b) identifying independent nets in said communication graph (col. 19 line 54 - col. 20 line 3, "When a graph 40 becomes very large, or a project is being worked on by several people, it will be natural to partition the task into several sub-graphs. These can be distinct graphs, with separate sources and sinks").

It would have been obvious to one of ordinary skill in the art to combine Hunt with French since in cases where a particular task is large, the time required to generate a minimum cost cut of the graph may prove to be prohibitively high. Thus, to modify Hunt with French would have been obvious in order to calculate minimum costs for smaller graphs, while maintaining data dependencies between the sub-graphs, such that all communication links are still intact.

As per claim 2, Hunt discloses a task management method as in claim 1, wherein the communication graph generated in step (a) comprises:

task components represented as nodes of said communication graph (col. 24 lines 8-28, "Create one node for each unit in the applications"); and

edges connecting ones of said nodes representing communication between connected nodes (col. 24 lines 8-28, "Create one edge between every pair of communication units. The weight on the edge should be the difference between communication cost [communication time] for the remote case [when two application units are placed on separate machines] and the local case [when the two application units are placed on the same machine]").

As per claim 3, Hunt discloses a task management method as in claim 2, after the step (a) of generating a communication graph, further comprising the steps of:

- a1) weighting edges, said edges being weighted proportional to communication between connected nodes (col. 24 lines 8-28, "The weight on the edge should be the difference between communication cost [communication time] for the remote case [when two application units are placed on separate machines] and the local case [when the two application units are placed on the same machine]"); and
- a2) assigning terminal nodes, task components being placed on said terminal nodes in the task placing step (d) (col. 24 lines 8-28, "Create two additional nodes: the source and the sink", wherein the source and sink are endpoints of the communication flow).

As per claim 4, French discloses a task management method as in claim 3, wherein the step (b) of identifying independent nets comprises the steps of:

- i) selecting a seed node for an independent net (col. 10 line 39 - col. 11 line 63, "Traversals of the graph 40 are initiated by an external change to a traversal context which is referenced from a root 43 of the graph 40");

Art Unit: 2127

- ii) identifying nodes adjacent to said seed node as perimeter nodes belonging to said independent net, perimeter nodes being an outer perimeter of nodes identified as belonging to said independent net (col. 10 line 39 - col. 11 line 63, "There are two ways to implement traversals of the graph. The first approach is to consider a conventional explicit depth-first traversal of the DAG", wherein the traversal continues by visiting the children of each node, i.e. adjacent nodes, recursively);
- iii) identifying nodes adjacent to said perimeter nodes as belonging to said independent net, said identified adjacent nodes being identified as perimeter nodes (col. 10 line 39 - col. 11 line 63, "the downstream connections are recursively traversed"); and
- iv) repeating step (iii) until all perimeter nodes are terminal nodes (col. 10 line 39 - col. 11 line 63, "the downstream connections are recursively traversed").

It would have been obvious to one of ordinary skill in the art to combine Hunt with French for reasons discussed above in reference to claim 1.

As per claim 9, Hunt discloses a task management method as in claim 3, wherein each said task component is a unit of the computer program (col. 24 lines 8-28, "Create one node for each unit in the application").

As per claim 10, Hunt discloses a task management method as in claim 9, wherein said each computer program unit is an instance of an object in an object-oriented program (col. 1 lines 36-55, "Various types of modular software, including software designed in an object-oriented framework, can conceivably be distributed throughout a distribution system", wherein

Art Unit: 2127

the placement of task components corresponds to the distribution of objects in an object oriented application).

As per claim 11, Hunt discloses a task management method as in claim 9, wherein in step (d) computer program units are placed on computers, computer program units being placed on a common computer being combined into a single component (col. 24 lines 42-48, "The MIN-CUT algorithm will cut through the edge that is least expensive...thus leaving the application unit attached to the computer on which its aggregate communication and computation time is the lowest").

As per claim 12, Hunt discloses a distributed processing system for determining optimal placement of computer program components on multiple computers, said distributed processing system comprising:

means for generating a communication graph representative of a computer program (col. 23 lines 13-23, "the application units and inter-unit communication form a commodity flow network", wherein the application units are components of a task or an application program, and the inter-unit communication provides information pertaining to the weighting of edges);

means for determining a min cut for the communication graph (col. 24 lines 8-28, "the algorithm to map a client-server distributed partitioning problem onto the MIN-CUT problem is as follows", wherein the algorithm for determining the minimum cut of the graph is disclosed);

means for placing program components on ones of multiple independent computers responsive to said min cut determined for the communication graph (col. 23 lines 13-23, "After

Art Unit: 2127

all data has been gathered, it is the optimization algorithm that decides where application units will be placed on the network”, wherein the optimization algorithm is a minimum cut algorithm and finds the paths of minimal communication costs); and

said computer program being executed by said multiple independent computers (Figs 1-2, wherein the computer program components are distributed among various nodes of a network and executed according to the minimum cut solution).

French discloses the following limitations not shown by Hunt, specifically:

means for identifying independent nets in said communication graph (col. 19 line 54 - col. 20 line 3, “When a graph 40 becomes very large, or a project is being worked on by several people, it will be natural to partition the task into several sub-graphs. These can be distinct graphs, with separate sources and sinks”).

It would have been obvious to one of ordinary skill in the art to combine Hunt with French for reasons discussed above in reference to claim 1.

As per claim 13, Hunt discloses a distributed processing system as in claim 12, wherein the communication graph comprises:

a plurality of nodes, each of said plurality of nodes representing one of said program components (col. 24 lines 8-28, “Create one node for each unit in the applications”); and

a plurality of edges connecting ones of said nodes, each of said edges representing communication between connected nodes (col. 24 lines 8-28, “Create one edge between every pair of communication units. The weight on the edge should be the difference between communication cost [communication time] for the remote case [when two application units are



Art Unit: 2127

placed on separate machines] and the local case [when the two application units are placed on the same machine]”).

As per claim 14, Hunt discloses a distributed processing system as in claim 13, further comprising:

weighting means for weighting said edges proportional to communication between connected said nodes (col. 24 lines 8-28, “The weight on the edge should be the difference between communication cost [communication time] for the remote case [when two application units are placed on separate machines] and the local case [when the two application units are placed on the same machine]”) and the communication graph further comprises terminal nodes, task components being placed on said terminal nodes (col. 24 lines 8-28, “Create two additional nodes: the source and the sink”, wherein the source and sink are endpoints of the communication flow).

As per claim 15, French discloses a distributed processing system as in claim 14, wherein the means for identifying independent nets comprises:

means for selecting a seed node amongst the nodes of said communication graph (col. 10 line 39 - col. 11 line 63, “Traversals of the graph 40 are initiated by an external change to a traversal context which is referenced from a root 43 of the graph 40”); and

means for branching out from said seed node and identifying perimeter nodes adjacent (col. 10 line 39 - col. 11 line 63, “There are two ways to implement traversals of the graph. The first approach is to consider a conventional explicit depth-first traversal of the DAG”, wherein

Art Unit: 2127

the traversal continues by visiting the children of each node, i.e. adjacent nodes, recursively) until all perimeter nodes are terminal nodes (col. 10 line 39 - col. 11 line 63, "the downstream connections are recursively traversed").

It would have been obvious to one of ordinary skill in the art to combine Hunt with French for reasons discussed above in reference to claim 1.

As per claims 21-23 and 27-28, Hunt discloses a computer program product for partitioning a graph, said computer program product comprising a computer usable medium having computer readable program code thereon, said computer readable program code implementing the method of claims 1-4, and 9-11, respectively (Fig. 2, wherein the method of distributing task components on a network is implemented in software intended for use in a computer system). The limitations of claims 21-23 and 27-28 are similar to those found in claims 1-4 and 9-11, respectively. Therefore, the discussion presented above is pertinent to the present claims as well.

3. Claims 5-7, 16-17, and 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunt in view of French in view of Dave (USPN 6,230,303).

As per claim 5, Dave discloses the following limitations not shown by the modified Hunt, specifically a task management method as in claim 4, wherein before the step (i) of selecting a seed node, all nodes not being terminal nodes are marked as unvisited nodes (col. 19 lines 10-37,

Art Unit: 2127

“Mark all tasks as unvisited”, wherein the clustering of tasks is performed in a graph traversal similar to that of French, and (1) in the clustering algorithm corresponds to the claimed step (i)).

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Dave since marking nodes as visited or unvisited allows the algorithm to easily determine if the traversal has completed, or if more nodes require processing. Although Dave is related to clustering of tasks such that power consumption is reduced, the method is applicable to reducing computation and communication time as well (col. 19 lines 15-17, “Clustering along the higher energy-level path makes the communication time as well as communication energy for inter-task edges zero”). Therefore, since both the methods of French and Dave are related to reducing the computation time for a large graph by dividing it into smaller sub-graphs, the methods are easily combinable.

As per claim 6, Dave discloses the following limitations not shown by the modified Hunt, specifically a task management method as in claim 5, wherein in step (i) the seed node is marked as visited and perimeter nodes are marked as visited in steps (ii) and (iii) (col. 19 lines 10-37, “If  $t_i$  is a sink task, ...[m]ark  $t_i$  as visited”, “If  $t_i$  is not a sink task, ...[m]ark  $t_i$  as visited”, wherein the since the sink is treated as the seed or root node, the marking of the sink as visited corresponds to step (ii) and the marking of the non-sink nodes as visited corresponds to step (iii)).

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Dave for reasons discussed above in reference to claim 5.

As per claim 7, Dave discloses the following limitations not shown by the modified Hunt, specifically a task management method as in claim 6, wherein all nodes marked as visited in steps (i) - (iv) identify an independent net, said method further comprising the steps of:

- v) checking said communication graph for unvisited nodes (col. 19 lines 10-37, "For each unvisited task  $t_i$  in the task graph, do the following:", wherein the step is repeated until there are no remaining unvisited nodes); and
- vi) repeating step (i) - (v) whenever unvisited nodes remain in said communication graph (col. 19 lines 10-37, "For each unvisited task  $t_i$  in the task graph, do the following:", wherein the step is repeated until there are no remaining unvisited nodes).

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Dave for reasons discussed above in reference to claim 5.

As per claim 16, Dave discloses the following limitations not shown by the modified Hunt, specifically a distributed processing system as in claim 15, further comprising means for marking nodes as unvisited nodes (col. 19 lines 10-37, "Mark all tasks as unvisited", wherein the clustering of tasks is performed in a graph traversal similar to that of French) or as visited nodes (col. 19 lines 10-37, "If  $t_i$  is a sink task, ...[m]ark  $t_i$  as visited", "If  $t_i$  is not a sink task, ...[m]ark  $t_i$  as visited").

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Dave for reasons discussed above in reference to claim 5.

As per claim 17, Dave discloses a distributed processing system as in claim 16, wherein the marking means marks each perimeter node as visited, when all nodes in said communication graph are marked as visited all independent nets have been identified in said communication graph (col. 19 lines 10-37, "For each unvisited task  $t_i$  in the task graph, do the following:", wherein the step of clustering is repeated until there are no remaining unvisited nodes).

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Dave for reasons discussed above in reference to claim 5.

As per claims 24-25, Hunt discloses a computer program product for partitioning a graph, said computer program product comprising a computer usable medium having computer readable program code thereon, said computer readable program code implementing the method of claims 5-7 (Fig. 2, wherein the method of distributing task components on a network is implemented in software intended for use in a computer system). The limitations of claims 24-25 are similar to those found in claims 5-7. Therefore, the discussion presented above is pertinent to the present claims as well.

4. Claims 8 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunt in view of French in view of Padberg et al. (cited by Applicant in IDS filed Jan. 3, 2001, paper number 3) (hereinafter Padberg).

As per claim 8, Padberg discloses the following limitations not shown by the modified Hunt, specifically a task management method as in claim 3, wherein the step (c) of determining a min cut comprises the steps of:

- i) listing all independent nets as subgraphs in a subgraph list (pg. 27-30, procedure SETUP, wherein the procedure computes the adjacency lists of each node, thereby identifying all independent nets in the graph);
- ii) selecting a subgraph from said subgraph list (pg. 27-30, procedure TEST1, wherein the step of choosing an edge from the adjacency list selects one of the subgraphs);
- iii) applying a linear complexity method to said subgraph, if said linear complexity method divides said subgraph into two or more smaller independent nets, listing said smaller independent nets in said subgraph list and returning to step (i) (pg. 27-30, procedure TEST1, wherein the complexity of the procedure is  $O[mn]$ , which is linear and the procedure MIN\_CUT is repeated as long as the cardinality of the graph is greater than one);
- iv) checking whether said subgraph includes two or more smaller independent nets, if said subgraph includes two or more smaller independent nets, identifying and listing said smaller independent nets in said subgraph list and returning to step (i) (pg. 27-30, procedure MIN\_CUT, wherein the procedure is repeated as long as the cardinality of the graph is greater than one);
- v) applying a higher complexity method to said subgraph, said higher complexity method being more complex than said linear complexity method and, if said higher

complexity method divides said subgraph into two or more smaller independent nets, listing said smaller independent nets in said subgraph list and returning to said step (i) (pg. 27-30, procedure TEST2, wherein the complexity of the procedure is  $O\{mn \log[n^2/m]\}$ , which is of higher complexity than the linear method and the procedure MIN\_CUT is repeated as long as the cardinality of the graph is greater than one);

vi) selectively collapsing an edge to reduce said subgraph, if collapsing said edge divides said subgraph into two or more smaller independent nets, listing said smaller independent nets in said subgraph list and returning to step (i) (pg. 27-30, procedure SHRINK, wherein multiple nodes are collapsed into one, such that the number of edges in the list are reduced); and

vii) checking whether said subgraph list is empty (pg. 26, "The basic idea is to repeatedly shrink subsets of nodes of the graph G that satisfy the sufficient conditions for shrinkability until a graph with only one node is obtained").

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Padberg since the procedure for determining a minimum cost cut of a graph is of great importance in terms of attempting to reduce communication time, particularly for network communication. However, many minimum cut procedures can be costly and difficult to implement, potentially negating the benefit gained from finding the minimum cut. The minimum cut procedure of Padberg provides an improvement upon previous minimum cut procedures in that it has the same worst case bound as previous methods, but it is easier to implement and has been shown to be more efficient than previous methods, especially in cases of graphs with a large number of nodes (see pages 31-35 for computational results).

As per claim 26, Hunt discloses a computer program product for partitioning a graph, said computer program product comprising a computer usable medium having computer readable program code thereon, said computer readable program code implementing the method of claim 8 (Fig. 2, wherein the method of distributing task components on a network is implemented in software intended for use in a computer system). The limitations of claim 26 are similar to those found in claim 8. Therefore, the discussion presented above is pertinent to the present claims as well.

5. Claims 18-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunt in view of French in view of Dave in view of Padberg.

As per claim 18, Padberg discloses the following limitations not shown by the modified Hunt, specifically a distributed processing system as in claim 17, wherein the means for determining a min cut comprises:

means for maintaining a list of all independent nets (pg. 27-30, procedure SETUP, wherein the procedure computes the adjacency lists of each node, thereby identifying all independent nets in the graph);

linear complexity reduction means for selectively reducing listed independent nets (pg. 27-30, procedure TEST1, wherein the complexity of the procedure is  $O[mn]$ , which is linear and the procedure MIN\_CUT is repeated as long as the cardinality of the graph is greater than one);



higher complexity reduction means for selectively reducing listed independent nets using a method having higher complexity than used by said linear complexity means (pg. 27-30, procedure TEST2, wherein the complexity of the procedure is  $O\{mn \log[n^2/m]\}$ , which is of higher complexity than the linear method and the procedure MIN\_CUT is repeated as long as the cardinality of the graph is greater than one);

means for selectively collapsing independent net edges to reduce said listed independent nets (pg. 27-30, procedure SHRINK, wherein multiple nodes are collapsed into one, such that the number of edges in the list are reduced);

means for checking whether any reduced independent net includes two or more smaller independent nets (pg. 27-30, procedure MIN\_CUT, wherein the procedure MIN\_CUT is repeated as long as the cardinality of the graph is greater than one); and

means for checking whether said list is empty (pg. 26, "The basic idea is to repeatedly shrink subsets of nodes of the graph G that satisfy the sufficient conditions for shrinkability until a graph with only one node is obtained").

It would have been obvious to one of ordinary skill in the art to combine the modified Hunt with Padberg since the procedure for determining a minimum cost cut of a graph is of great importance in terms of attempting to reduce communication time, particularly for network communication. However, many minimum cut procedures can be costly and difficult to implement, potentially negating the benefit gained from finding the minimum cut. The minimum cut procedure of Padberg provides an improvement upon previous minimum cut procedures in that it has the same worst case bound as previous methods, but it easier to

Art Unit: 2127

implement and has been shown to be more efficient than previous methods, especially in cases of graphs with a large number of nodes (see pages 31-35 for computational results).

As per claim 19, Hunt discloses a distributed processing system as in claim 18, wherein each said program component is a unit of the computer program (col. 24 lines 8-28, "Create one node for each unit in the application").

As per claim 20, Hunt discloses a distributed processing system as in claim 19, wherein said each program unit is an instance of an object in an object-oriented program (col. 1 lines 36-55, "Various types of modular software, including software designed in an object-oriented framework, can conceivably be distributed throughout a distribution system", wherein the placement of task components corresponds to the distribution of objects in an object oriented application).

### ***Conclusion***

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The additional cited references pertain to partitioning of task graphs in order to reduce complexity, and of representing a task being scheduled within directed graphs.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

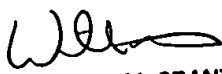
Art Unit: 2127

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William A Grant can be reached on (703) 308-1108. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.



Syed Ali  
December 2, 2003



WILLIAM GRANT  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100  
12/9/03